

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



**IMPLEMENTAÇÃO DE ARQUITETURA
PARA BASE APLICACIONAL EM J2EE**

Jaime Mota Vaz

PROJETO

PÚBLICA

MESTRADO EM ENGENHARIA INFORMÁTICA
Especialização em Especialização em Arquitetura, Sistemas
e Redes de Computadores

2012

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



IMPLEMENTAÇÃO DE ARQUITETURA PARA BASE APLICACIONAL EM J2EE

Jaime Mota Vaz

PROJETO

Trabalho orientado pelo Prof. Doutor Carlos Alberto Pacheco dos Anjos Duarte

e co-orientado por Ana Teresa Raimundo de Almeida

MESTRADO EM ENGENHARIA INFORMÁTICA

Especialização em Especialização em Arquitetura, Sistemas e
Redes de Computadores

2012

Agradecimentos

Este relatório marca o final do meu percurso académico, e quero agradecer desde já à Faculdade de Ciências pela excelente formação que me proporcionou.

A todos os meus colegas de curso um muito obrigado pelo vosso companheirismo ao longo destes anos. Quero fazer um especial agradecimento aos meus colegas Pedro Pereira, Igor Antunes e ao Carlos Cândido que além de me acompanharem durante todo o meu percurso académico, souberam apoiar-me nos momentos mais difíceis.

Quero agradecer também a todos os meus colegas de trabalho, pela excelente receção, integração e companheirismo revelados durante o estágio. Um especial agradecimento ao Nuno Reis por todos os conhecimentos que me transmitiu e pelo seu apoio prestado na realização deste relatório. Sem esquecer a Teresa Almeida, quero gratificá-la pela sua excelente orientação e pela oportunidade que me proporcionou de trabalhar com excelentes profissionais.

Ao meu coordenador de estágio, Professor Carlos Duarte, agradeço toda a sua disponibilidade e apoio, não só na escrita deste relatório como também ao longo de todo o percurso académico. Durante a minha formação, este foi incansável a transmitir a sua sabedoria e a angariar excelentes oportunidades para os alunos, demonstrando sempre o seu ótimo profissionalismo.

Por último, quero agradecer a toda a minha família, por todo o apoio, a motivação, o carinho e o auxílio que me deram desde sempre, mas sobretudo nesta etapa da minha vida académica.

Para vocês Mãe e Luísa
Obrigado pelo vosso amor e apoio

Resumo

O projeto descrito neste relatório teve como objetivo criar uma plataforma que fosse modular e uma base para a criação de aplicações *Rich Internet Application* empresariais.

A abordagem para implementação desta plataforma assentou em 3 grandes blocos: Interface para interação com o utilizador, Content Management System (CMS) e Business Process Management (BPM). O trabalho realizado neste projeto centrou-se principalmente nos componentes de CMS e BPM. O CMS é um sistema de gestão de conteúdos que será utilizado para simplificar a atualização de conteúdos na plataforma e o BPM é um motor de gestão de processos de negócio.

Esta plataforma foi utilizada para realizar um protótipo funcional, em que se criou uma aplicação CRM (*Customer Relationship Management*) para a qual foram desenvolvidos alguns módulos que explorassem as potencialidades da plataforma.

Esta plataforma despertou interesse num cliente da Accenture, e a mesma foi utilizada para efetuar uma aplicação à medida deste. Durante a implementação desta plataforma foi adiada a integração do CMS e do BPM para uma próxima iteração, o que impossibilitou a implementação da parte correspondente no contexto do estágio. Como o cliente em questão utiliza um *middleware* para acesso aos dados de negócio, foi necessário criar uma camada que obtivesse e gerisse os dados. Além disso, foi criado um módulo de links que vai de acordo com as necessidades do cliente.

O desenvolvimento deste projeto permitiu alcançar todos os objetivos delineados com sucesso.

Palavras-chave: Gestão de conteúdos, Regras de Negócio, Gestão de clientes, J2EE, Arquitetura orientada a serviços

Abstract

The project described in this report aims to create at creating a modular platform that is a base for creating business Rich Internet Applications.

The implementation approach for this platform is based on three main blocks: Interface for user interaction, Content Management System (CMS) and Business Process Management (BPM). The work focused mainly on the CMS and BPM components. CMS is a content management system that will be used to simplify updating content on the platform. BPM is a business process management engine.

This platform was used to develop a functional prototype, with which a CRM (Customer Relationship Management) application was created, for which some modules have been developed that exploit the potential of the platform.

This platform has sparked interest in one of Accenture's clients, and, as a result, it was used to make an application oriented to his business. During the implementation of this platform the integration of CMS and BPM was delayed to the next iteration, making it impossible to implement those in the context of the internship. As the customer in question uses a middleware for accessing business data, it was necessary to create a layer to obtain and managed all the data. Moreover, a module of user links was built according to the needs of the customer to integrate it in the application created.

The development of this project has achieved all objectives outlined successfully.

Keywords: Content Management System, Business Process Management, Customer Account Management, J2EE, Service Oriented Architecture

Conteúdo

Capítulo 1	Introdução.....	1
1.1	Motivação	1
1.2	Objetivos.....	3
1.3	Organização do documento	4
Capítulo 2	Instituição de acolhimento.....	7
2.1	Instituição e equipa de trabalho	7
2.2	Metodologia de Trabalho.....	8
2.3	Ferramentas de desenvolvimento	10
Capítulo 3	Análise e Desenho	11
3.1	Enquadramento	11
3.2	Ambiente de Execução	12
3.3	Descrição da arquitetura técnica.....	12
Capítulo 4	Plataforma construída	13
4.1	Layout aplicacional.....	13
4.2	Conectores e Handlers CMS e BPM	13
4.3	Instalação e análise do Alfresco	13
4.4	Integração e análise do Activiti	14
4.5	Filtro de Logs	14
4.6	Módulo de campanhas	14
4.7	Módulo de pesquisa e acesso de documentos.....	14
4.8	Módulo de notas do cliente.....	14
Capítulo 5	Implementação da plataforma em cliente.....	15
5.1	Enquadramento	15
5.2	Integração SOA	16
5.2.1	Introdução ao problema.....	16
5.2.2	Requisitos	20
5.2.3	Modelação	21
5.2.4	Testes.....	21

5.3	Módulo de Links Úteis	21
5.3.1	Contextualização do módulo	21
5.3.2	Origem dos dados	21
5.3.3	Modelação do módulo de configuração de <i>links</i>	22
5.3.4	Modelação do módulo de apresentação de <i>links</i>	22
5.3.5	Testes unitários	22
5.4	Outras tarefas realizadas	22
5.4.1	Testes de sistema	22
5.4.2	Plano de instalação	23
5.4.3	Análise para distribuição de carga	23
Capítulo 6	Discussão	25
6.1	Síntese do projeto desenvolvido	25
6.2	Planeamento	26
6.3	Sumário de todas tarefas concretizadas	27
6.4	Conclusões e trabalho futuro	28
Bibliografia		31

Lista de Figuras

- Fig. 1 - Estrutura interna da Accenture
- Fig. 2 - Modelo *V-model*
- Fig. 3 - [Imagem confidencial]
- Fig. 4 - [Imagem confidencial]
- Fig. 5 - [Imagem confidencial]
- Fig. 6 - [Imagem confidencial]
- Fig. 7 - [Imagem confidencial]
- Fig. 8 - [Imagem confidencial]
- Fig. 9 - [Imagem confidencial]
- Fig. 10 - [Imagem confidencial]
- Fig. 11 - [Imagem confidencial]
- Fig. 12 - [Imagem confidencial]
- Fig. 13 - [Imagem confidencial]
- Fig. 14 - [Imagem confidencial]
- Fig. 15 - [Imagem confidencial]
- Fig. 16 - [Imagem confidencial]
- Fig. 17 - [Imagem confidencial]
- Fig. 18 - [Imagem confidencial]
- Fig. 19 - Arquitetura Tibco Information bus
- Fig. 20 - Representação Envelope SOAP
- Fig. 21 - Modelo WSDL
- Fig. 22 - [Imagem confidencial]
- Fig. 23 - [Imagem confidencial]
- Fig. 24 - [Imagem confidencial]
- Fig. 25 - [Imagem confidencial]
- Fig. 26 - [Imagem confidencial]
- Fig. 27 - [Imagem confidencial]
- Fig. 28 - [Imagem confidencial]
- Fig. 29 - [Imagem confidencial]
- Fig. 30 - [Imagem confidencial]

Lista de Tabelas

Tabela 1 – Planeamento das atividades a desenvolver durante o estágio

Tabela 2 – Tarefas realizadas

Lista de Acrónimos

AD - Active Directory
ADS - Accenture Delivery Suite
AJAX - Asynchronous Javascript And XML
AOP - Aspect-Oriented Programming
API - Application Programming Interface
BPM - Business Process Management
BPMN2 - Business Process Modeling Notation V.2
CMIS - Content Management Interoperability Services
CMS - Content Management System
CRM - Customer Relationship Management
DAO - Data Access Object
ECM - Enterprise Content Management
HTML - Hypertext Markup Language
HTTP - Hypertext Transfer Protocol
IDE - Integrated Development Environment
J2EE - Java 2 Platform, Enterprise Edition
JAXB - Java Architecture For XML)
JSP - Javaserer Pages
MVC - Model-View-Controller
POJO - Plain Old Java Object
POM - Project Object Model
RIA - Rich Internet Application
SOA - Service Oriented Architecture
SOAP - Simple Object Access Protocol
STS - Springsource Tool Suite
SVN - Subversion
UML - Unified Modeling Language
URL - Uniform Resource Locator
WSDL - Webservice Description Language
WS-I - Webservices Interoperability Organization
XML - Extensible Markup Language

Capítulo 1

Introdução

Este relatório de estágio descreve um projeto realizado na Accenture Portugal, enquadrado no âmbito da disciplina de Projeto de Engenharia Informática, que tem como objetivos mitigar alguns problemas de grandes empresas e, pessoalmente, concluir o Mestrado em Engenharia Informática, da Faculdade de Ciências da Universidade de Lisboa.

1.1 Motivação

A gestão de grandes empresas é bastante complexa, e com o evoluir do seu volume de negócios, estas vão-se deparando com problemas que podem ser ultrapassados com a utilização de tecnologia. De seguida são enumerados alguns dos problemas com que as empresas se encaram diariamente.

Gestão documental, colaboração e gestão de conteúdos

Quando uma empresa alcança uma dimensão considerável, poderão ocorrer problemas que afetam o controlo da empresa e também o desempenho de funções dos colaboradores. Por exemplo, a inexistência de um arquivo digital pode traduzir-se num enorme problema, pois dificulta a colaboração entre funcionários, e poderá criar problemas de duplicação de documentos e de controlo de versões. Além disso, tem um impacto na segurança da empresa pois os documentos físicos podem-se perder ou fotocopiar com relativa facilidade e o seu extravio poderá acarretar enormes prejuízos tanto intelectuais como financeiros.

Outro problema comum às empresas é a disponibilização e manutenção de conteúdos *online*, pois tipicamente estas alterações requerem que as mesmas sejam efetuadas por pessoas qualificadas em desenvolvimento Web.

Uniformização e controlo de processos de negócio

Numa empresa podem existir inúmeros processos de negócio. Em processos complexos é comum existirem atividades que só devem ser executadas por um conjunto de pessoas. A decisão de quem deve efetuar a próxima iteração do processo pode ser determinado por um conjunto de variáveis, que nem sempre são de acesso direto. A determinação do fluxo, sendo uma tarefa manual, poderá originar falhas humanas no seguimento correto do processo. A passagem do mesmo para uma equipa errada implica atrasos, ou uma terminação errónea de um processo, ou, num caso mais extremo, pode originar fugas de informação.

Cooperação entre aplicações

Nas empresas é comum existirem um grande conjunto de ferramentas que foram construídas e utilizadas ao longo dos anos de funcionamento destas. A necessidade de colocar essas aplicações a colaborar com outras ferramentas torna-se um problema complicado porque nem sempre as tecnologias permitem uma interação direta.

Dispersão de informação empresarial

Quando o desenvolvimento das empresas é orientado para competir em novas frentes de negócio, os novos conhecimentos adquiridos, tendencialmente, vão ser armazenados em novas localizações. Quando um colaborador necessita de lidar com várias áreas de negócio, torna-se complicado gerir as referências para as fontes que lhe são mais relevantes para o seu dia-a-dia. Este fator é muito importante para os colaboradores obterem uma ótima rentabilidade.

Outro problema que as empresas têm é no controlo da informação obsoleta e em direcionar os colaboradores para as informações atualizadas. Por exemplo, se existir um determinado produto que foi descontinuado, o colaborador irá permanecer com a mesma referência para o produto, sem que seja mostrado um produto equivalente.

Ao limitar o impacto destes problemas é possível reduzir custos relativos a logística de documentos, aumentar a produtividade, a eficiência dos recursos humanos, assim como aumentar a qualidade do serviço prestado ao cliente na medida em que

todos os processos ficam unificados e supervisionados por responsáveis. Cooperando com os sistemas existentes é possível inovar mantendo a retrocompatibilidade dos mesmos.

1.2 Objetivos

Face aos problemas expostos na motivação é necessário desenvolver soluções que possam contribuir para a mitigação dos mesmos.

No problema da gestão documental, colaboração e gestão de conteúdos a solução passa por utilizar um gestor de conteúdos. Com um gestor de conteúdos é possível manter todo o arquivo centralizado, manter um controlo de cópias e versões em tempo real, podendo também definir-se permissões de acesso a documentos. Os gestores de conteúdos podem de igual modo ser utilizados para efetuar gestão de conteúdos disponibilizados *online*, podendo os mesmos serem alterados de uma forma muito simples, evitando o uso de pessoas qualificadas em desenvolvimentos Web, obtendo assim meios para efetuar uma manutenção regular de conteúdos.

Para se obter uma uniformização e controlo de processos de negócio, uma solução passa por implementar uma ferramenta de gestão de processos de negócio, de forma a manter os processos uniformes e transversais em toda a empresa. Com um sistema deste tipo, a distribuição de tarefas é realizada pelo mesmo, fazendo com que este guie o utilizador nas suas tarefas. Com a ajuda deste tipo de ferramentas, torna-se possível a otimização dos processos de negócio, assim como aumentar a produtividade, melhorar a utilização dos recursos, aumentar a flexibilidade e a adaptação à mudança.

Para resolver o problema de cooperação entre aplicações, uma solução é utilizar uma arquitetura orientada a serviços, em que são trocadas mensagens em formato texto onde são abstraídas as particularidades dos diversos sistemas.

A dispersão de informação empresarial poderá ser ultrapassada se existir uma ferramenta que proporcione o armazenamento, disponibilização e configuração dessas referências. Esta solução beneficia o colaborador na medida em que são disponibilizados meios para realizar as suas tarefas de uma forma rápida e intuitiva. No entanto beneficia também a empresa, na medida em que esta consegue obter um

controlo das origens da informação e torna transparente uma necessidade de mobilização de conteúdos para o colaborador.

Este projeto teve como objetivo construir uma plataforma recorrendo a frameworks *open-source* que incluía um gestor de conteúdos e um motor de processos de negócio, que seja modular, extensível, sólida, adaptável a todo o tipo de negócio, intuitiva, personalizável, rápida e de alta disponibilidade, disponibilizando uma ótima solução para utilização no mercado empresarial. Esta plataforma vai simplificar o acesso a todas as tecnologias num único *front-end*, sendo uma solução *all in one* que oferece a possibilidade de desenhar uma aplicação à medida para o cliente.

1.3 Organização do documento

Este documento é constituído por 6 capítulos, e pretende apresentar o conteúdo e o produto desenvolvido durante o estágio.

O primeiro capítulo tem como finalidade introduzir o leitor ao contexto do trabalho e apresentar os objetivos do mesmo, bem como oferecer uma introdução ao produto desenvolvido.

O segundo capítulo apresenta a entidade onde decorreu o estágio, a sua metodologia de trabalho e as ferramentas que utilizam para o desenvolvimento aplicacional.

O terceiro capítulo apresenta os requisitos da aplicação, uma análise para o ambiente aplicacional e uma descrição da arquitetura técnica utilizada para a construção da plataforma.

O quarto capítulo apresenta a plataforma que foi construída, apresentando o tipo de *layout*, uma análise de funcionalidades do CMS e do BPM e a realização de um filtro de *logs*. É apresentado também um protótipo funcional de uma aplicação CRM, construído no âmbito do estágio, em que foram realizados alguns módulos para integrar na plataforma.

O quinto capítulo apresenta uma integração da plataforma num cliente, em que foi realizada uma aplicação à medida, onde existiu a necessidade de realizar ajustes na arquitetura para concretizar a integração de um *middleware* que se expõe via *Webservices* para obter os dados de negócio do cliente. Esta integração foi realizada utilizando uma metodologia de desenvolvimento de componentes, de forma a

possibilitar o reuso do componente noutras aplicações futuras. Foi realizado um módulo de *links* úteis para integrar na plataforma, com o objetivo de facilitar o acesso a informações de negócio aos colaboradores do cliente.

Por fim, no sexto capítulo apresenta-se uma síntese do trabalho desenvolvido, o planeamento que estava previsto para o estágio, um sumário de todas as tarefas concretizadas e o trabalho futuro que poderia ser realizado.

Capítulo 2

Instituição de acolhimento

2.1 Instituição e equipa de trabalho

A Accenture é uma organização global de serviços de consultoria de gestão, tecnologias de informação e *outsourcing*. Foi fundada em 1989, como Andersen Consulting, e tem cerca de 236000 colaboradores em todo o mundo.

A estratégia da Accenture baseia-se no seu conhecimento e na sua experiência para oferecer serviços. A procura pelas mais recentes tecnologias faz com que esta consiga identificar novos negócios e tendências. A Accenture tem a capacidade de desenvolver soluções para ajudar os seus clientes a encontrar novos mercados de negócios, aumentar as receitas e melhorar o desempenho operacional, tornando os seus processos mais eficientes.

A Accenture está distribuída em 5 grupos operacionais:

- *Communications, Media & High Tech*: engloba comunicações, eletrónica, média e entretenimento;
- *Financial services*: abrange o sector da banca, o mercado de capitais e seguradoras;
- *Resources*: engloba o sector de energias, recursos naturais, e químicos;
- *Health & public service*: abrange o sector da saúde e serviços públicos;
- *Products*: engloba todos os produtos que não se enquadrem nos grupos referidos anteriormente. Este grupo abrange o sector automóvel, bens de consumo, serviços industriais, transportes, entre outros.

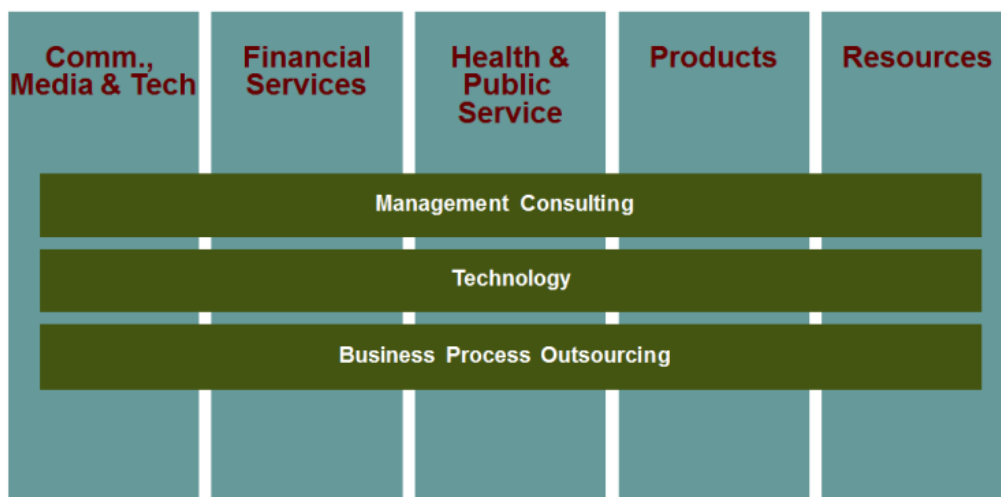


Fig. 1 - Estrutura interna da Accenture

Estes 5 grupos estão ainda divididos em 3 áreas sendo estas, consultoria de gestão, tecnologia e *outsourcing*. Esta estrutura está representada na Fig.1.

Durante o estágio fui inserido numa equipa que pertencia ao grupo de *Communications, Media & High Tech* na área de *Technology*. Este grupo era composto por 11 elementos, estando estes hierarquizados por funções e responsabilidades. Esta hierarquia tem como principais níveis:

1. *Manager*;
2. *System Analyst*;
3. *Programmer*.

Durante o decorrer do estágio fui orientado por todos os colegas, existindo, no entanto, dois elementos que estavam diretamente responsáveis pela minha orientação no projeto, sendo estes um *manager* e um *system analyst*.

Durante a implementação do projeto num cliente, foi-me dada a possibilidade de participar em 3 reuniões e também o contacto com a equipa de produção.

2.2 Metodologia de Trabalho

No âmbito do projeto foi utilizada a metodologia proprietária da Accenture para desenvolvimentos de soluções à medida.

A metodologia de projeto fundamenta-se nas melhores práticas e contempla as fases de planeamento, análise, desenho, construção, testes e manutenção.

A metodologia especifica métodos e procedimentos para arquitetura técnica, formação, suporte para um aumento de performance e operacionalização de serviços.

Esta metodologia de ciclo de vida de projeto pode ser equiparada à metodologia *V-Model*. As fases deste modelo estão representadas na Fig. 2:

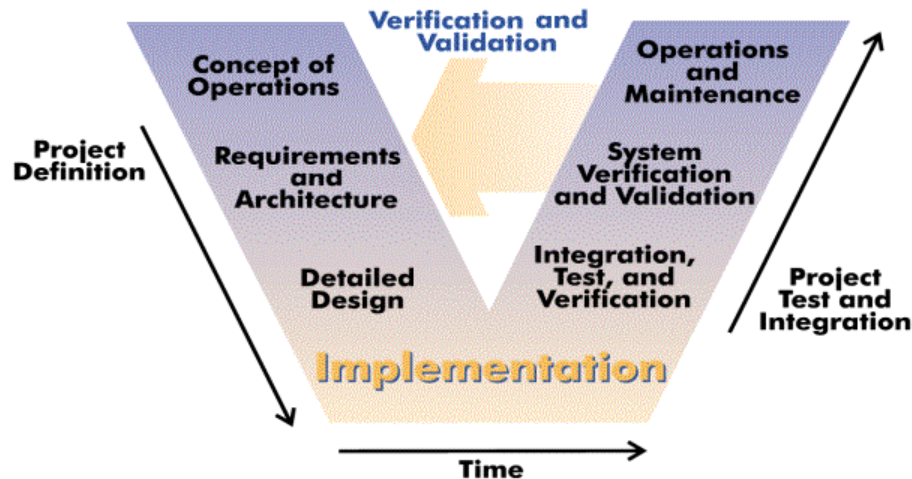


Fig. 2 - Modelo *V-model* [34]

A grande diferença neste modelo, para um modelo em cascata, é que na análise do problema são definidos em detalhe os casos de teste. Cada iteração de definição do *V-model* resulta um plano de testes que, posteriormente na fase de integração, será utilizado para efetuar os testes à aplicação. Deste modelo resultam 4 planos de testes:

- Testes unitários;
- Testes de integração;
- Testes de sistema;
- Testes de aceitação.

Os testes unitários são feitos ao mesmo tempo que é desenvolvida a aplicação e são da responsabilidade do desenvolvedor. Este teste assegura que um módulo desenvolvido está de acordo com os requisitos. Para a realização deste teste o desenvolvedor deve realizar um conjunto de testes que garantam a integridade de todos os passos do algoritmo realizado e que os resultados obtidos são iguais aos esperados.

Os testes de integração são da responsabilidade do analista de sistemas. Neste teste são integrados todos os módulos da aplicação e são verificados os requisitos funcionais, a integridade, o desempenho e o conteúdo.

Os testes de sistema são da responsabilidade do analista de testes, e o propósito destes é testar a aplicação como um todo e realizar testes de recuperação de falhas, segurança, *stress* e *performance*.

Os testes de aceitação são executados por parte do cliente, para avaliar se a aplicação está de acordo com os seus requisitos.

A utilização deste modelo de ciclo de vida minimiza os custos de correção de erros porque quanto mais tarde o erro for detetado, maiores são os custos que estão inerentes à sua correção.

Devido à dimensão do projeto, o desenvolvimento foi efetuado em equipa, tendo sido utilizado um controlo de versões para armazenar e distribuir o projeto. Foram conduzidas algumas reuniões, conforme a necessidade de organização, ou discussão de assuntos relativos ao mesmo.

2.3 Ferramentas de desenvolvimento

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial estão descritas todas as ferramentas utilizadas pela organização no desenvolvimento do projeto descrito.

Capítulo 3

Análise e Desenho

Este capítulo aborda a análise e o desenho da plataforma posteriormente desenvolvida: os seus requisitos, algumas das suas funcionalidades, e toda a arquitetura por detrás da mesma.

3.1 Enquadramento

Como apresentado anteriormente a plataforma a construir deverá ser uma plataforma dinâmica e modular, que se pode ajustar a todos os tipos de organizações e negócios.

Esta deverá integrar um gestor de conteúdos que permita realizar todo o tipo de gestão documental e de conteúdos. Um gestor de conteúdos possibilita a realização de um grande leque de aplicações que necessite envolver documentos, conteúdos e colaboração. Com a utilização de um gestor de conteúdos é também possível oferecer a possibilidade de fazer alterações na plataforma de uma forma intuitiva e simples, para que estas possam ser efetuadas por colaboradores com conhecimentos básicos em informática.

Com a inclusão de um sistema de gestão de processos de negócios será possível definir fluxos de trabalho, onde os utilizadores da plataforma são guiados na realização das tarefas, garantindo assim uma uniformização de processos transversal a toda a organização, mitigando a existência de erros que poderão ocorrer em processos manuais. Com este sistema será também possível manter um controlo de quem efetua as tarefas, mantendo um registo de todos os processos realizados na plataforma.

A restante secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial, a mesma expõe os requisitos da aplicação e as funcionalidades pretendidas.

3.2 Ambiente de Execução

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial é exposta o tipo de aplicação desenvolvida e o ambiente de execução que foi utilizado.

3.3 Descrição da arquitetura técnica

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial está descrita a arquitetura técnica do projeto. Esta arquitetura é constituída por várias camadas que separam as diferentes responsabilidades operacionais. Todas as camadas são descritas no relatório confidencial na sua subsecção correspondente.

Capítulo 4

Plataforma construída

Durante os primeiros meses de estágio foi montada a plataforma descrita na análise. Foi elaborado um protótipo funcional num contexto de uma aplicação de CRM (*Customer Relationship Management*), implementando as funcionalidades já descritas na análise. Integrou-se na aplicação uma base de dados com dados de negócio já existentes na Accenture para realização de testes e também um *Active Directory* para gestão de acessos à aplicação.

4.1 Layout aplicacional

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial é descrito o tipo de *layout* aplicacional e a sua distribuição.

4.2 Conectores e Handlers CMS e BPM

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial são descritos os conectores e os *handler* concebidos para gerir a ligação e comunicação com o CMS e BPM.

4.3 Instalação e análise do Alfresco

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se descrita toda a análise de características e funcionalidades efetuada ao CMS Alfresco.

4.4 Integração e análise do Activiti

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se descrita toda a análise de características e funcionalidades efetuada ao BPM Activiti.

4.5 Filtro de Logs

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se exposto um componente realizado para integrar na plataforma para realizar uma filtragem de *logs*.

4.6 Módulo de campanhas

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se descrito um módulo de campanhas efetuado para integrar na plataforma.

4.7 Módulo de pesquisa e acesso de documentos

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se exposto um módulo de pesquisa e acesso de documentos realizado para integrar na plataforma.

4.8 Módulo de notas do cliente

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se descrito um módulo de notas de cliente concretizado para integrar na plataforma.

Capítulo 5

Implementação da plataforma em cliente

Este capítulo vai refletir o trabalho desenvolvido na integração da plataforma no cliente. Este está dividido em 4 subcapítulos. O primeiro enquadra esta implementação, no segundo aborda-se a integração com os *Webservices*, no terceiro apresenta-se um módulo que foi realizado para adicionar ao layout da aplicação e, por fim, um quarto subcapítulo descreve outras tarefas realizadas durante o projeto.

5.1 Enquadramento

Durante a implementação da plataforma surgiu a oportunidade de implementar a mesma num cliente. Inicialmente foi elaborada uma proposta para a plataforma integrar o CMS e o BPM. No entanto, e após a análise de alguns protótipos, o cliente colocou de parte a inclusão do CMS e do BPM para uma redução de riscos e diminuição do tempo de desenvolvimento do projeto, ficando estas funcionalidades em análise para uma próxima iteração.

No cliente encontrava-se implementado um *middleware* da Tibco para acesso aos dados de negócio do cliente. Este *middleware* faz a exposição de dados através de *Webservices*, tendo, por isso, sido necessário concretizar uma solução que efetuasse a comunicação com os mesmos.

O restante da secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se expostas as alterações realizadas na arquitetura de forma a implementar a *framework* no cliente.

5.2 Integração SOA

Neste subcapítulo vai ser abordada a integração dos *Webservices* na aplicação desenvolvida. Inicialmente é feita uma breve introdução ao problema, seguido de um estudo realizado sobre alguns conceitos, uma análise de requisitos e o desenvolvimento do componente.

5.2.1 Introdução ao problema

Toda a informação de negócio da aplicação a desenvolver vai ser exposta via *Webservices*. Este foi um requisito do cliente pois este possui uma camada de integração implementada no seu negócio. O cliente forneceu algumas APIs que já se encontravam em exploração, e outras, novas, foram criadas para preencher os requisitos do projeto.

A camada de integração existente é uma plataforma que foi fornecida pela Tibco, sendo esta responsável pela criação e gestão de *Webservices* para aquisição ou alteração de dados.

Para utilizar este tipo de tecnologia foi necessário modelar um módulo para integrar a utilização deste *middleware* na aplicação. Segundo as boas práticas de desenvolvimento da Accenture, o módulo foi construído numa perspetiva de ser um componente, para futuramente utilizar o mesmo noutros projetos, onde exista a necessidade de integração com este *middleware*.

De seguida vão ser abordados alguns dos temas cruciais, que foram alvo de estudo para a realização deste componente.

Tibco

A Tibco *Software*[19] é um fornecedor de soluções orientado aos negócios. Os produtos Tibco possibilitam a distribuição de informações em tempo real através de uma tecnologia denominada por Information Bus™ (Fig. 19).

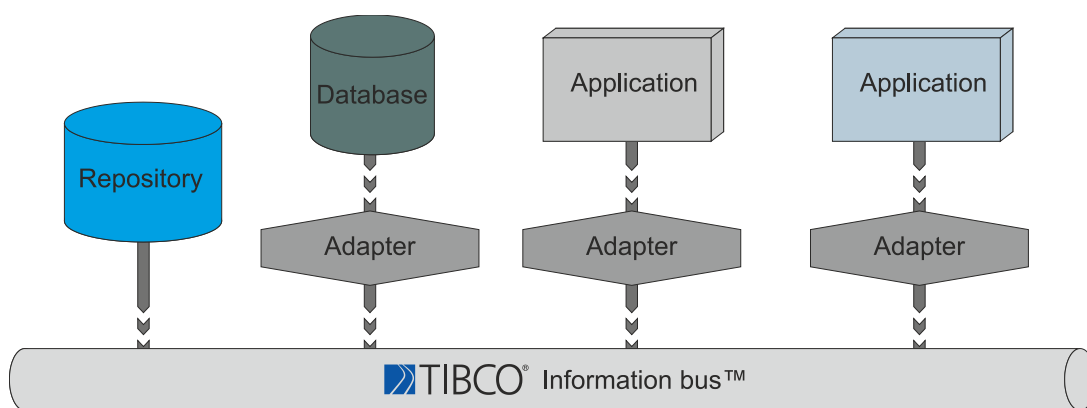


Fig. 19 - Arquitetura TIBCO Information bus

O TIB Middleware é um *software* que faz uma ponte de comunicação entre serviços utilizando Information Bus™. Este suporta três tipos distintos de interação entre aplicações, que podem ser:

- Pedido/resposta *unicast*, como, por exemplo, uma consulta ou transação a um servidor específico;
- Pedido/resposta *broadcast*, como, por exemplo, consultas que podem originar várias respostas de um ou mais servidores;
- *Publish/subscribe*, como a distribuição de informações para os consumidores associados.

Os benefícios desta arquitetura são inúmeros[20], como, por exemplo:

- Facilidade em criar novos subsistemas – adicionar novos serviços sem necessidade de mudar código de um subsistema;
- Facilidade em mover os sistemas – como, por exemplo, para outras localizações geográficas sem alterar os subsistemas, permitindo assim implementar também uma tolerância a falhas onde outro sistema pode assumir o sistema que falhou;
- Facilita o desenvolvimento – os desenvolvedores utilizam uma *API* própria para a comunicação, não necessitando assim de projetar as ligações;
- Facilita a mudança – é possível mudar, por exemplo, de base dados, mantendo todos os serviços a funcionar, sem que sejam necessárias alterações aos sistemas existentes;
- Fácil de monitorizar através de aplicações proprietárias.

No contexto deste projeto será utilizada a ferramenta Tibco enquanto solução de integração na medida em que esta já se encontra em exploração no cliente. Dos serviços que se encontravam implementados no cliente, foram reutilizados os necessários e criados novos serviços para o acesso à informação de negócio. A criação dos novos serviços não foi da minha responsabilidade, estando portanto fora do âmbito deste relatório.

Software baseado em componentes

Diariamente as aplicações vão se tornando cada vez mais complexas, e devido aos requisitos do mercado, sistemas de alta qualidade têm de ser construídos num curto espaço de tempo com o menor custo possível. O *software* baseado em componentes é um paradigma que enfatiza o desenho e a construção de sistemas usando componentes de *software* reutilizáveis. Num desenho de uma arquitetura utilizando este tipo de construção, ao invés de efetuar uma análise das tarefas num nível mais detalhado, são examinados os requisitos para determinar qual o subconjunto que é diretamente suscetível de composição em vez de executar uma construção de raiz. Este modelo de desenvolvimento traz inúmeros benefícios como a redução do tempo de desenvolvimento, redução no custo do projeto, um aumento de produtividade e a reutilização de componentes robustos, o que se traduz num consequente aumento de qualidade no trabalho desenvolvido[35].

Um componente é uma unidade complexa, independente, que pode ser facilmente substituído devido ao seu baixo acoplamento para realizar a sua função. Um componente contém interfaces bem definidas e documentadas, e deve ser independente do contexto da aplicação assim como do sistema operativo onde opera [32].

SOAP Simple Object Access Protocol

SOAP é um protocolo de comunicação para troca de mensagens estruturadas em formato XML entre aplicações, que são enviadas tipicamente através do protocolo HTTP, sendo, no entanto, possível utilizá-lo em qualquer protocolo. O SOAP tem a grande vantagem de ser independente da plataforma, da linguagem da programação, do modelo de objetos, sendo por isso um protocolo muito utilizado para comunicação entre aplicações. Além disso, o protocolo HTTP é suportado por todos os *browsers* e servidores e o SOAP acaba por ser o protocolo mais utilizado para comunicações entre clientes e *Webservices*. O protocolo SOAP é uma abstração de um envelope, que

envolve as mensagens XML. De seguida encontra-se uma representação do envelope (Fig. 20).

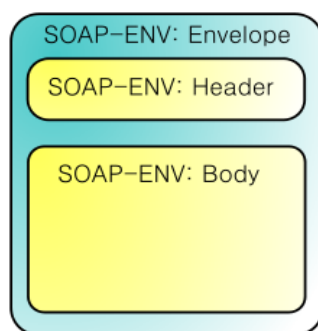


Fig. 20 - Representação Envelope SOAP

O envelope é composto por um cabeçalho e um corpo. O cabeçalho tipicamente contém informações sobre a aplicação, por exemplo informações relativas à autenticação. O corpo da mensagem contém a mensagem enviada pela aplicação.

Arquitetura orientada a serviços (SOA)

SOA (*Service-oriented architecture*) é um paradigma que está relacionado com a arquitetura corporativa, que permite a criação de serviços de negócio interoperáveis que podem facilmente ser reutilizados/partilhados entre aplicações e empresas [27].

Numa arquitetura orientada a serviços, partimos do pressuposto que os serviços são reutilizáveis, ou seja, a implementação do serviço é genérica o suficiente para ser utilizada noutros projetos. Os serviços partilham um contrato formal, onde estão descritas todas as funcionalidades do serviço, sendo este denominado por WSDL (*Webservice Description Language*). Um ficheiro WSDL é responsável por identificar o endereço no qual um serviço está publicado, o protocolo e os seus parâmetros de entrada e saída. O ficheiro WSDL é composto por várias secções :

- *Definition* – Este é o elemento *root* de todos os elementos de um WSDL. Este define o nome do *Webservice*, contém as declarações dos *namespaces* e contém todos os elementos descritos abaixo;
- *Data Types* – Descrição dos tipos de dados que serão trocados no serviço em si. Ex. *String* – nome da variável;
- *Message* – Descrição dos dados que são transmitidos tais como os parâmetros de *input* e de *output*;

- *Port Type* – Descreve a operação ou método que vai ser chamado, assim como o tipo de operação que é chamada. O mais comum é o *Request-Response* onde um pedido requer uma resposta, sendo este o utilizado na aplicação;
- *Binding* – Descreve o protocolo de como os dados são enviados; tipicamente são descritas as ordens de como os dados são transmitidos;
- *Service* – Descreve a localização do serviço[26].

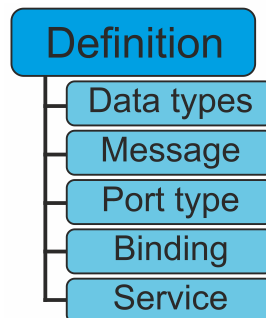


Fig. 21 - Modelo WSDL

Os serviços têm um acoplamento muito baixo entre eles, ou seja, cada serviço é independente para realizar as suas tarefas. Estes devem ser tratados como uma caixa preta, e a sua programação poderá ser substituída em qualquer altura, sem influenciar aqueles que o utilizam.

Uma grande vantagem de utilização de serviços é a sua interoperabilidade. Esta vantagem provém principalmente de uma organização denominada por *Webservices Interoperability Organization (WS-I)*, que estabelece alguns *standards* para facilitar esta interoperabilidade entre plataformas, sistemas operativos e linguagens de programação [28].

A troca de mensagens é feita utilizando o protocolo SOAP que já foi abordado anteriormente.

5.2.2 Requisitos

Esta secção foi omitida devido à natureza confidencial deste relatório. Esta expõe os requisitos da modelação dos componentes realizados.

5.2.3 Modelação

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se expostos os componentes realizados no âmbito da integração no cliente.

5.2.4 Testes

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se descritos os testes realizados a este módulo e os resultados dos testes de carga

5.3 Módulo de Links Úteis

Neste subcapítulo vai ser abordado o módulo que foi realizado para a disponibilização de *links* úteis ao utilizador. Vai ser feita uma pequena contextualização do módulo, a origem dos dados para o preenchimento do módulo, como foi efetuada a modelação do mesmo e os testes que foram realizados após o desenvolvimento.

5.3.1 Contextualização do módulo

O cliente alvo da aplicação a desenvolver, tem um conjunto de aplicações Web e, por vezes, para responder aos seus clientes, pode ser necessário consultar fontes externas à aplicação. O módulo a desenvolver vem responder a essas necessidades, disponibilizando uma lista de endereços aos utilizadores da aplicação, para simplificar o acesso a determinadas aplicações de forma a obterem acesso às informações desejadas.

Sendo a aplicação uma ferramenta transversal aos colaboradores da empresa, que pertencem a diferentes ramos de negócio, os *links* disponibilizados acabam por não ter utilidade para todos os utilizadores, tendo por isso sido necessário construir um módulo de configuração que permitisse aos utilizadores escolherem e organizarem os *links* ao seu gosto. Caso estes não configurem os *links*, é apresentada uma lista por omissão.

5.3.2 Origem dos dados

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se descritos vários cenários da origem dos dados para o módulo construído.

5.3.3 Modelação do módulo de configuração de *links*

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se exposto um módulo construído para ser integrado na aplicação do cliente.

5.3.4 Modelação do módulo de apresentação de *links*

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontra-se exposto um módulo construído para ser integrado na aplicação do cliente.

5.3.5 Testes unitários

Esta secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se expostos os testes unitários realizados aos módulos.

5.4 Outras tarefas realizadas

Neste subcapítulo são abordadas outras tarefas que foram realizadas durante o estágio, que ainda não foram referidas neste relatório de estágio.

5.4.1 Testes de sistema

No final da implementação da aplicação integrei a equipa de testes onde efetuei testes de sistema. Nestes testes a aplicação é instalada num ambiente idêntico ao de produção, e esta é testada para garantir que esta cumpre com os requisitos propostos. A execução destes testes consistia na execução de 270 casos de testes para cada perfil de utilizador, que foram desenhados pela nossa equipa de análise funcional e pela equipa de *customer services* do cliente.

Os testes realizados permitiram encontrar alguns erros, que foram corrigidos, alguns por mim, outros pela restante equipa, dependendo da complexidade ou da origem do código. A correção dos erros é sempre mais rápida se for efetuada pelas pessoas que desenvolveram porque estas estão familiarizadas com o código. Esta abordagem permitiu uma resolução muito mais rápida do problema.

5.4.2 Plano de instalação

O plano de instalação foi da minha responsabilidade. Este plano tem como propósito ajudar as equipas de produção a realizarem as instalações no ambiente de produção. Este documento está organizado por passos, que devem ser seguidos pelas equipas de forma a concluírem a instalação com sucesso.

A restante secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se expostas informações relativas ao ambiente de execução do projeto.

5.4.3 Análise para distribuição de carga

A restante secção foi omitida devido à natureza confidencial deste relatório. Na versão confidencial encontram-se expostas informações relativas a infraestruturas do cliente.

Capítulo 6

Discussão

6.1 Síntese do projeto desenvolvido

Partes deste texto foram omitidas a fim de garantir a não divulgação de informações confidenciais, estando essas entre parênteses rectos.

Neste estágio foi realizada uma plataforma utilizando J2EE onde foi integrado um sistema de gestão de conteúdos e um motor de regras de negócio que permite implementar *Rich Web Applications*. [...] Foram analisadas as funcionalidades oferecidas pelo CMS e pelo BPM para a integração no portal.

Foi construído um protótipo funcional de uma aplicação de *Customer Relationship Management* em que foram adicionadas funcionalidades de pesquisa, visualização e edição de dados de cliente, assim como visualização de documentos, campanhas e notas. Para ir de encontro às funcionalidades propostas foi realizado um módulo de campanhas, um módulo de pesquisa e obtenção de documentos e um módulo de notas de cliente.

O protótipo desenvolvido suscitou o interesse por parte de um cliente, que originou a implementação de uma aplicação à sua medida.

A aplicação desenvolvida necessitou de alguns ajustes na arquitetura. Foi removido o CMS e o BPM numa primeira fase do projeto, de forma a minimizar os riscos e o tempo de entrega do projeto. Foram efetuadas alterações na arquitetura devido aos dados de negócio serem obtidos a partir de um *middleware*, em que este faz a sua exposição de dados através de *Webservices*. Esta integração suscitou alguns problemas de implementação com o [...], e para tal foi necessário desenhar alguns módulos para a

comunicação com o *middleware*, e para efetuar a receção e armazenamento dos dados de negócio no contexto da aplicação.

Foi construído um módulo de *links* para integrar na plataforma. Este foi inicialmente desenvolvido como prova de conceito, elaborado de forma a utilizar o CMS para armazenamento e gestão dos *links*. O módulo mais tarde teve de ser alterado devido à remoção do CMS, tendo sido os dados migrados para base de dados. O módulo foi construído de forma a estar de acordo com as necessidades do cliente.

6.2 Planeamento

O planeamento de atividades a desenvolver durante o estágio era o seguinte:

<i>Atividades</i>	<i>Mês</i>
Enquadramento funcional e técnico do projeto	1
Setup do ambiente de desenvolvimento	1
Setup da <i>framework</i> : Plataforma de CMS e BPM, Integração na solução	1,2
Produção do manual de instalação	2,3
Desenho detalhado do modelo operacional de desenvolvimento com gestão e controlo de versão	3,4
Desenvolvimento e testes do modelo operacional	5,6,7,8
Elaboração do manual da solução	9

Tabela 1 – Planeamento das atividades a desenvolver durante o estágio

O planeamento proposto teve de ser ajustado devido ao CMS e o BPM ser um dos temas principais do estágio e como estas duas tecnologias ficaram de fora do âmbito da implementação do projeto no cliente, a minha responsabilidade recaiu na realização dos componentes de acesso a dados de negócio e de um módulo de configuração de *links*. As tarefas efetivamente concretizadas podem ser consultadas na próxima secção.

6.3 Sumário de todas tarefas concretizadas

Em suma, as tarefas concretizadas podem ser sintetizadas através da seguinte tabela:

Etapa	Tarefas realizadas
Enquadramento funcional e técnico do projeto	Apresentação do contexto e requisitos do projeto
	Estudo das <i>frameworks</i> propostas para o projeto
Setup do ambiente de desenvolvimento	Estas tarefas foram omitidas devido à natureza confidencial deste relatório. São referidas tarefas relativas à instalação do ambiente de desenvolvimento
Setup da <i>framework</i> : Plataforma de CMS, Integração na solução	Instalação do pacote da <i>framework</i> Alfresco num servidor
	Teste de integração com diferentes tipos de base de dados
	Teste de integração com Active Directory
	Teste de atribuição de permissões
	Estudo de funcionalidades e limitações
	Abertura de documentos/ficheiros
	Criação de tarefas e <i>workflows</i> no CMS
	Criação de <i>links</i> e <i>datalists</i>
Setup da <i>framework</i> : Plataforma de BPM, Integração na solução	Instalação do pacote da <i>framework</i> Activiti num servidor
	Instalação do <i>plug-in</i> de criação de processos
	Estudo de funcionalidades e limitações
	Criação de uns processos exemplo
	<i>Deploy</i> de processos para o Activiti engine
	<i>Tracking</i> do estado do processo
Realização da Aplicação para protótipo funcional	Instalação e realização de uma biblioteca para integrar o Alfresco no projeto
	Construção de uma biblioteca para a integração do Activiti no projeto

	Elaboração de filtros para os <i>logs</i> para simplificar a deteção de erros
	Construção de módulo de campanhas
	Construção de módulo de pesquisa e acesso de documentos
	Construção de módulo de notas do cliente
Realização da Aplicação no cliente	Análise e desenho de solução para utilização do <i>middleware</i> da Tibco
	Realização dos componentes para obtenção de dados de negócio
	Realização do módulo de configuração de <i>Links</i>
	Realização do módulo de apresentação de <i>Links</i>
	Plano de instalação
	Testes integrados

Tabela 2 – Tarefas realizadas

6.4 Conclusões e trabalho futuro

Na realização deste estágio foi possível adquirir experiência profissional, integrar o mundo do trabalho e obter novos conhecimentos tanto de negócio como tecnológicos.

Este estágio permitiu angariar um leque variado de conhecimentos nas área de gestão de conteúdos, regras de negócio, J2EE, arquiteturas orientadas a serviços e aplicações RIA.

O Alfresco como solução de CMS é uma boa opção. No entanto, a versão *open-source* Community apresenta algumas restrições com configurações de base de dados e a documentação disponibilizada é um pouco escassa. Relativamente às bases de dados, esta apenas permite um leque restrito das mesmas, suportando apenas de origem o MySQL e PostgreSQL. Em relação à documentação, esta deixa muito a desejar porque a única disponibilizada encontra-se na wiki proprietária e é prestada pelos utilizadores da comunidade num fórum. A implementação deste sistema num ambiente de produção poderá trazer alguns dissabores. Caso seja necessário fazer alguma recuperação profunda do sistema, ou se ocorrer algum erro mais crítico que afete a execução normal do sistema, não é fácil arranjar uma solução em tempo útil sem recorrer ao apoio empresarial do Alfresco. Por isso, esta solução necessita de se aliar a uma subscrição

empresarial, ou então serão necessários mais esforços de forma a angariar conhecimentos de arquitetura e de código fonte base do Alfresco para obter alguma independência do apoio empresarial.

O Activiti demonstrou ser um motor de BPM muito dinâmico, completo, extremamente rápido e ao mesmo tempo sólido. Ao contrário do Alfresco, a documentação do Activiti é muito rica e objetiva, facilitando a utilização da plataforma. Com esta foi possível concretizar todos os objetivos pretendidos, e mesmo assim demonstrou que não se encontrava no limite das funcionalidades que era possível oferecer com o seu pacote inicial.

Em relação ao projeto realizado, considerou-se que a metodologia de trabalho estava adequada. No entanto, existiram alguns problemas com repetições de código, na medida que foram construídos métodos idênticos por vários programadores, devido a algumas falhas de documentação. O Javadoc deveria ter sido programado para ser compilado utilizando uma rotina, disponibilizando as APIs do código Java e Javascript num servidor para consulta interna. Esta medida iria certamente motivar os programadores a manter um Javadoc completo, atualizado e coerente.

A arquitetura proposta demonstrou ser sólida, modular e flexível, permitindo um rápido desenvolvimento e integração no cliente. Em cerca de 4 meses, foi possível entregar a aplicação ao cliente, mesmo tendo a necessidade de realizar os módulos para comunicação com o *middleware* para a obtenção dos dados de negócio. Esta aplicação agradou na generalidade todos os colaboradores do cliente, e o feedback destes demonstrou que a aplicação encontrava-se muito atrativa, com uma ótima usabilidade e rápida.

Numa próxima iteração pretende-se implementar toda a gestão da aplicação por gestor de conteúdos e os processos de negócio de acordo com as necessidades do cliente. Relativamente aos processos de negócio, seria interessante modular uma solução focada ao utilizador final, que de algum modo disponibilizasse uma ferramenta que permitisse um colaborador da área de negócios construir os fluxos de negócio necessários, sem ser necessário recorrer à equipa de desenvolvimento.

Em suma, foi construída uma boa base para desenvolvimentos de aplicações J2EE que traz muito valor e oportunidades de negócio lucrativas para a Accenture.

Bibliografia

- [1] Item bibliográfico omitido devido a motivos de confidencialidade.
- [2] Kiczales, G., Lamping, J. , Mendhekar, A., Maeda, C. , Lopes, C.V., Loingtier, J. & Irwin, J. in *Aspect-Oriented Programming*, Published in proceedings of the European Conference on Object-Oriented Programming (ECOOP), Finland. Springer-Verlag LNCS 1241. June 1997.
- [3] Item bibliográfico omitido devido a motivos de confidencialidade.
- [4] <http://mbanagouro.net/site/2012/02/09/introducao-ao-asp-net-mvc/>
- [5] Potts, J., in *Alfresco Developer Guide*, Packt Publishing, 2008
- [6] Shariff, M., Bhandari, A. , Choudhary, V. & Majumdar, P., in *Alfresco 3 Enterprise Content Management Implementation*, Packt Publishing, 2nd edition, 2009
- [7] Gestão Documental e Workflow com Alfresco, Fernando Fernandez, MoreData, PPT em <http://www.esop.pt/uploads/2009/05/sessao2-alfresco.pdf>
- [8] http://wiki.alfresco.com/wiki/Main_Page
- [9] <http://www.alfresco.com/about/awards/>
- [10] <http://www.slideshare.net/alfresco/alfresco-as-sharepoint-alternative-architecture-overview-presentation>
- [11] <http://www.slideshare.net/alfresco/webinar-slides-total-cost-of-ownership-for-ecm-ian-howells-alfresco-presentation>
- [12] Item bibliográfico omitido devido a motivos de confidencialidade.
- [13] http://wiki.alfresco.com/wiki/Alfresco_Repository_Architecture
- [14] http://en.q-bpm.org/mediawiki/index.php/Main_Page
- [15] http://en.q-bpm.org/mediawiki/index.php/Process_Model
- [16] <http://www.activiti.org/faq.html>
- [17] <http://www.activiti.org/components.html>
- [18] <http://www.activiti.org/userguide/index.html>
- [19] <http://www.Tibco.com>

- [20] <http://www.pbti.com.br/files/IntegracaodeProcessos.pdf>
- [21] Item bibliográfico omitido devido a motivos de confidencialidade.
- [22] Item bibliográfico omitido devido a motivos de confidencialidade.
- [23] Item bibliográfico omitido devido a motivos de confidencialidade.
- [24] Collins-Sussman, B. , Fitzpatrick, B. W., Pilato, C. M., in *Version Control with Subversion: For Subversion 1.7*, compilation r4275, Open Book 2002-2011
- [25] Item bibliográfico omitido devido a motivos de confidencialidade.
- [26] http://www.tutorialspoint.com/wsdl/wsdl_elements.htm
- [27] Erl, Thomas in *SOA: principles of service design*, Pearson Education, Inc., 2007
- [28] <http://www.ws-i.org/about/Default.aspx>
- [29] Item bibliográfico omitido devido a motivos de confidencialidade.
- [30] <http://www.oracle.com/technetwork/articles/javase/index-140168.html>
- [31] Item bibliográfico omitido devido a motivos de confidencialidade.
- [32] Larman, Craig in *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, 3rd Edition, Pearson Education, 2005
- [33] Item bibliográfico omitido devido a motivos de confidencialidade.
- [34] Clarus Concept of Operations. Publication No. FHWA-JPO-05-072, Federal Highway Administration (FHWA), 2005
- [35] George T. Heineman, William T. Councill in *Component-Based Software Engineering: Putting the Pieces Together*, Addison-Wesley Professional; 1st ed., 2001
- [36] Item bibliográfico omitido devido a motivos de confidencialidade.
- [37] Item bibliográfico omitido devido a motivos de confidencialidade.